

CLAIMS

This listing of claims will replace all prior listings of claims in the present application.

Listing of Claims:

1 (currently amended) A method for displaying an animation, comprising:

receiving an instruction to display an animation, the animation comprising a plurality of displayable images ordered for sequential display;

retrieving an animation file responsive to the instruction, the animation file providing an ordering of the images;

determining a maximum size, the maximum size related to a maximum amount of memory usable to load images;

determining a first set of the images, ~~which in display order, which~~ aggregate to a size ~~less than a~~ up to the maximum size, the first set of images having a final image;

determining a second set of the images, ~~which in display order, which~~ aggregate to a size ~~less than a~~ up to the maximum size, ~~an image~~ the images in the second set being in sequence behind the final image;

generating a first segment file indicative of the first set of images, the first set of images not linked to each other;

generating a second segment file indicative of the second set of images, the second set of images not linked to each other;

associating a callback identifier with the second segment file;

providing the callback identifier along with the first segment file;

loading the first set of images into ~~an animation processor memory~~ readable by an animation engine according to the first segment file;

displaying sequentially, using the image order in the animation file, each image in the first set as a first animation segment;

retrieving the callback identifier from the first segment file;

using the callback identifier to load the second set of images into the ~~animation processor~~ memory according to the second segment file; and

displaying sequentially, using the image order in the animation file, each image in the second set as a second animation segment.

2 (original) The method for displaying an animation according to claim 1, wherein each of the images are stored as individual graphics files.

3 (original) The method for displaying an animation according to claim 1, wherein the animation file further comprises information indicative of the size of individual ones of the images, and the size information is used in determining the first set of images.

4 (currently amended) The method for displaying an animation according to claim 1, wherein the maximum size is set ~~to~~ to further correspond to a number of images of known size.

5 (currently amended) The method for displaying an animation according to claim 1, wherein the maximum size is set at a predetermined memory size for an embedded system.

6 (original) The method for displaying an animation according to claim 1, wherein the maximum size is generated responsive to an inquiry regarding available memory.

7 (original) The method for displaying an animation according to claim 1, wherein the first segment file provides a file identifier for each of the images in the first set.

8 (original) The method for displaying an animation according to claim 1 wherein the associating step includes using the callback identifier as a name for the second segment file.

9 (original) The method for displaying an animation according to claim 1 wherein the associating step includes placing the callback identifier as data in the second segment file.

10 (original) The method for displaying an animation according to claim 1 wherein providing the callback identifier includes using the callback identifier as part of a name for the first segment file.

11 (original) The method for displaying an animation according to claim 1 wherein providing the callback identifier includes placing the callback identifier as data in the first segment file.

12 (original) The method for displaying an animation according to claim 1 wherein the second set of images are being loaded into the animation processor while the images in the first set are being displayed.

13 (currently amended) A method of generating animation segment files, comprising:
receiving an animation file that identifies and orders a set of displayable images;
determining a maximum size, the maximum size related to a maximum amount of
memory usable to load images;
dividing the set of images into sequential subsets of images, each subset having a
size up to a maximum size, where individual images have no link to another image, being
~~smaller than a maximum size and~~ indicative of an animation segment;
associating a subset identifier with each respective subset;
associating an action instruction with each respective segment; and
wherein the action instruction associated with one subset identifies another one of
the subsets.

14 (currently amended) The method of generating animation segment files according to claim 13, wherein the maximum size ~~is set~~ further corresponds to a number of images of known size.

15 (currently amended) The method of generating animation segment files according to claim 13, wherein the maximum size is set to a memory size derived from an amount currently usable for loading images.

16 (original) The method of generating animation segment files according to claim 13, wherein an action instruction is used to identify the last subset.

17 (currently amended) A method of displaying an animation, comprising:

- receiving an instruction to display the animation, the animation comprising a set of sequential displayable images;

- retrieving a first segment file, the first segment file identifying a first subset of the images, the first subset of images not linked to each other;

- loading the first subset of images and sequentially displaying the images in the first subset of images;

- extracting a callback instruction using the first segment file, the callback instruction identifying a second segment file;

- releasing memory holding at least one of the images in the first subset of images;

- retrieving the second segment file, the second segment file identifying a second subset of the images, the second set of images not linked to each other; and

- loading the second subset of images and sequentially displaying the images in the second subset of images.

18 (original) The method of displaying an animation according to claim 17, wherein loading the second subset is initiated before all the images in the first subset have been displayed.

19 (canceled)

20 (currently amended) A set of animation segment files in computer readable format comprising:

- a first segment file on a computer readable medium and storing a first subset of displayable images, the images not linked to each other;

- a callback instruction associated with the first segment file;

a second segment file on the computer readable medium and storing a second subset of displayable images, the images not linked to each other;

a file identifier associated with the second segment file; and

wherein the callback instruction is indicative of the file identifier, the first segment loadable to a memory readable by an animation engine and the second segment loadable to the memory after at least a portion of the memory used by the first segment has been designated as available for loading.

21 (currently amended) The set of animation segment files according to claim 20, further comprising:

a third segment file on the computer readable medium and storing a third subset of displayable images;

an action instruction associated with the third segment file; and

wherein the action instruction indicates the third segment file is the last segment file.

22 (original) The set of animation segment files according to claim 20, further including a graphics file storing the first subset of images and the second subset of images.

23 (original) The set of animation segment files according to claim 20, further including a plurality of graphics file storing the first subset of images and the second subset of images.

24 (currently amended) A method for ~~sequencing~~ enabling concurrent presentation of a plurality of media objects into a presentation, comprising:

receiving a media file providing ~~an ordering of the media objects so that a first one of the media objects is presented before and a second one of the media objects, the~~ second media object being an animation file;

associating a callback identifier with the second media object ;

providing the callback identifier along with the first media object;

loading the first media object into a ~~media processor~~ memory usable for
presenting the first media object;
retrieving the callback identifier that was provided along with the first media
object;
using the callback identifier to load the second media object into ~~the media~~
~~processor memory~~; and memory usable for presenting the second media object
~~presenting the second media object~~ where the animation file has at least a first
and second segment associated with it, each segment comprising at least one image, each
image being a displayable image, the first segment identifying a first set of images not
linked to each other if there are a plurality of images, the second segment identifying a
second subset of images not linked to each other if there are a plurality of images, where
each segment has a size up to a maximum size, the maximum size being related to a
maximum amount of memory usable for loading images.

25 (currently amended) The method ~~for sequencing according to~~ of claim 24, wherein at
least one of the media objects is a sound file and at least another one of the media objects
is an animation file the first media object is a sound file.

26-27 (canceled)

28 (currently amended) The method ~~for sequencing according to~~ of claim 24, further
including a third one of the media objects, the third media object having an action
instruction indicative of a duration to present the third media object.

29 (currently amended) The method ~~for sequencing according to~~ of claim 24, wherein the
first media object has an action instruction for loading and ~~presenting~~ a third one of the
media objects, the third media object ~~being presented~~ enabled for presentation
concurrently with the first media object.

30 (currently amended) The method ~~for sequencing according to~~ of claim 29, wherein the
third media object ~~continues to be presented~~ is enabled for presentation after the second

media object ~~has started to be presented.~~